

Subspace-Aware Index Codes

Bhavya Kailkhura^{*†}, *Member, IEEE*,

Lakshmi Narasimhan Theagarajan^{*‡}, *Member, IEEE*,

Pramod K. Varshney[‡], *Fellow, IEEE*

Abstract

In this paper, we generalize the well-known index coding problem to exploit the structure in the source-data to improve system throughput. In many applications (e.g., multimedia), the data to be transmitted may lie (or can be well approximated) in a low-dimensional subspace. We exploit this low-dimensional structure of the data using an algebraic framework to solve the index coding problem (referred to as *subspace-aware index coding*) as opposed to the traditional index coding problem which is *subspace-unaware*. Also, we propose an efficient algorithm based on the alternating minimization approach to obtain near optimal index codes for both subspace-aware and -unaware cases. Our simulations indicate that under certain conditions, a significant throughput gain (about 90%) can be achieved by subspace-aware index codes over conventional subspace-unaware index codes.

Keywords – *Index coding, coded side-information, low-dimensional data, alternating minimization*

I. INTRODUCTION

Index coding with side-information (ICSI) [1], [2], [3], is a problem, where a server has N stored messages that it can broadcast over a noiseless channel to a set of receivers or clients. Each client has a subset of the N messages as side information, and requests a subset of messages that it needs from the server. The objective of the ICSI problem is to devise an optimal coding strategy that minimizes the number of broadcast transmissions made by the server to satisfy the requirements of all the clients. The optimality criterion of an index code is its code length. The optimal index code length, i.e., the minimum number of transmissions required from the server for successful recovery of the desired information at the clients, was first characterized in [1],

This work was supported in part by NSF Grant no. ECCS 1609916.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-JRNL-718227

^{*}These authors contributed equally to this work.

[†] This author is presently affiliated to Lawrence Livermore National Laboratory, kailkhura1@llnl.gov

[‡]These authors are at the Department of EECS, Syracuse University, New York, {ltheagar, varshney}@syr.edu.

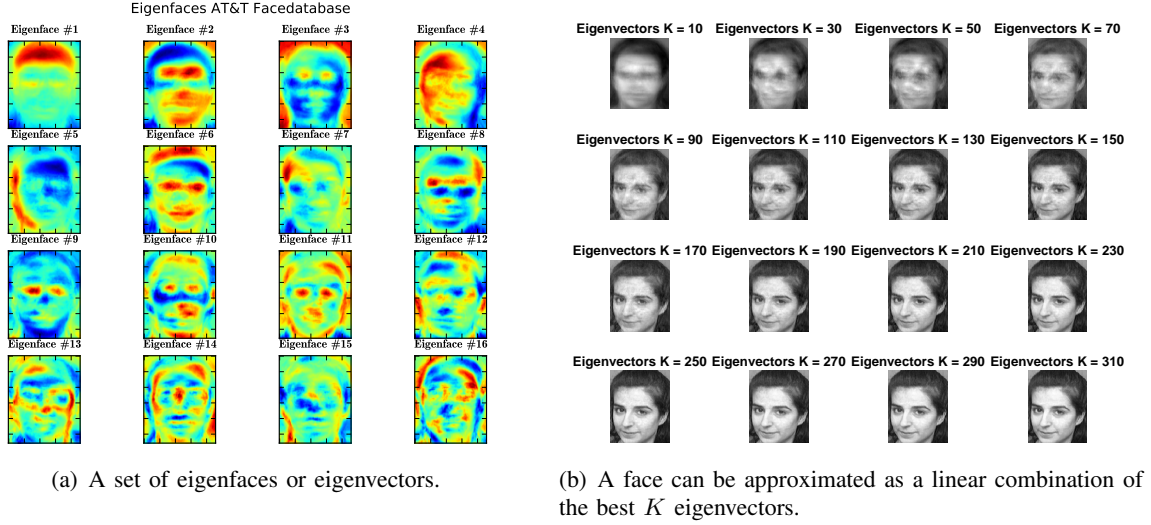


Fig. 1. Illustration of practical scenarios where data lies in a low dimensional linear subspace.

[2] as the minimum rank of a matrix that represents the side-information graph [2]. A method to construct index codes by solving a matrix completion problem was presented in [4]. In many practical scenarios, users possess coded side information (CSI) [5]; and index codes for linear CSI were studied in [5], [6]. Index codes over real field and their construction methods are investigated in [7], [8]. Considering index codes over the real field enables the use of efficient optimization techniques to construct near optimal index codes. Further, it was shown in [9] that network codes are equivalent to index codes. Thus, one can construct an optimal network code by constructing an optimal index code for the equivalent problem. Network codes over real field are discussed in [10], [11].

The source-data encountered in many practical systems such as data caching, images, video streaming, big-data storage and processing, can be well approximated using a lower dimensional linear subspace [12], [13], [14] (and references therein). Motivated by such applications, we propose a technique to construct index codes when the source-data belongs to a lower dimensional linear subspace. For example, it is well known that a facial image is a point from a high-dimensional image space which can be well approximated in a lower-dimensional linear subspace. The lower-dimensional subspace is found using Principal Component Analysis, which identifies the axes with maximum variance. In Figure 1(a), a set of eigenvectors (known as eigenfaces) are shown for AT&T Facedatabase. There are ten different images of each of the 40 distinct subjects. The size of each image is 92x112 pixels, with 256 grey levels per pixel. In Figure 1(b),

we can see that a good reconstruction quality can be obtained using a very small number of eigenfaces.¹

More specifically, the main contributions of this paper can be summarized as follows.

- We generalize the index coding problem with coded (and/or uncoded) side information to exploit the low-dimensional structure that may be present in the source-data.
- We establish bounds on the gain achieved by subspace-aware index codes over subspace-unaware case.
- We consider the design of subspace-aware/unaware index codes with coded/uncoded side information in a unified optimization framework and develop an efficient algorithm to construct near optimal index codes.
- Finally, we provide theoretical guarantees and simulation results on the performance of the proposed techniques.

The notations followed in the rest of this paper are: $\text{rk}(\cdot)$ denotes the rank of a matrix, $\text{span}(\cdot)$ denotes a vector space spanned by a set of vectors, $(\cdot)^\dagger$ denotes the pseudo-inverse of a matrix, and $\|\cdot\|_F$ denotes the Frobenius norm of a matrix.

II. PROBLEM SETUP

Consider a network with U users and a data source (DS). Let N denote the total number of data packets involved in a transmission instance, P denote the size of each data packet, \mathbf{x}_i denote the data in the i th packet, $\mathbf{x}_i \in \mathbb{R}^P$ for $i = 1, 2, \dots, N$, and $\mathbf{x} \triangleq [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{PN}$. The j th user requests V_j number of packets from the DS, \mathcal{R}_j is the set of all indices of the requested data packets by the j th user, $|\mathcal{R}_j| = V_j$ for $j = 1, 2, \dots, U$, and $\mathbf{x}_{\mathcal{R}_j}$ denotes the $PV_j \times 1$ information vector requested by the j th user. Each user possesses a linearly coded side information. Let M_j denote the length of the CSI and $\mathbf{S}_j \in \mathbb{R}^{PM_j \times PN}$ denote the side information coding matrix for the user j , $0 \leq M_j < N$. The CSI of the j th user is given by the vector $\mathbf{S}_j \mathbf{x}$. When the j th user has uncoded side information (USI), the side information consists of M_j data packets, and the non-zero columns of \mathbf{S}_j form an identity matrix of dimension $PM_j \times PM_j$.

If the vector \mathbf{x} belongs to a low-dimensional subspace, then $\mathbf{x} = \mathbf{T}\mathbf{w}$, where $\mathbf{T} \in \mathbb{R}^{PN \times PD}$ ($1 \leq D < N$) is the matrix of basis vectors of the low-dimensional subspace, $\mathbf{w} \in \mathbb{R}^{PD}$, and $\text{rk}(\mathbf{T}) = PD$.

¹These results are obtained from <https://github.com/bytefish/facerec>.

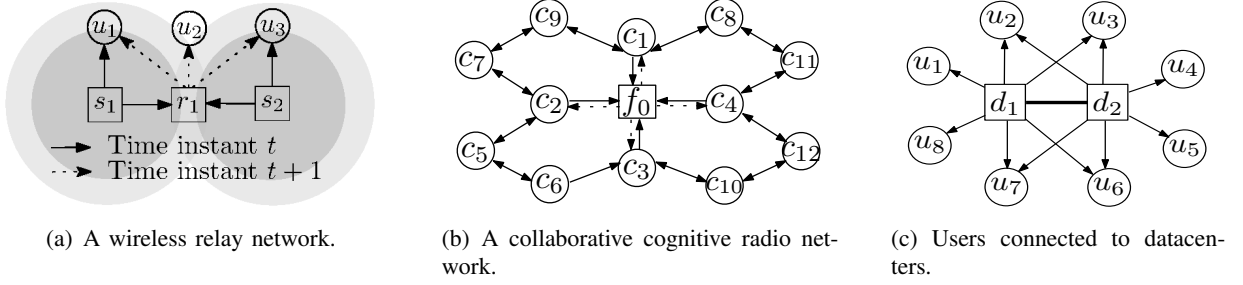


Fig. 2. Illustration of practical scenarios where index codes can be employed to improve efficiency and throughput.

Goal: Knowing \mathcal{R}_j , matrices \mathbf{S}_j and subspace structure \mathbf{T} for $j = 1, 2, \dots, U$, the goal is to have the DS broadcast the least number of coded data packets to U users such that each user is able to successfully decode the requested packets. \square

Let $\mathbf{y} \triangleq [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_L]^T \in \mathbb{R}^{PL}$ be the data vector transmitted by the DS. Now, each user needs to decode $\mathbf{x}_{\mathcal{R}_j} \in \mathbb{R}^{PV_j}$ from $[(\mathbf{S}_j \mathbf{x})^T \mathbf{y}^T]^T$. Assuming linear decoding, the j th user performs the decoding as $\hat{\mathbf{x}}_{\mathcal{R}_j} = \mathbf{D}_j [(\mathbf{S}_j \mathbf{x})^T \mathbf{y}^T]^T$, where \mathbf{D}_j is the decoding matrix. For linear encoding, this problem can be stated as follows.

Problem: Find a matrix $\mathbf{C} \in \mathbb{R}^{PL \times PN}$ such that

$$\mathbf{D}_j \begin{bmatrix} \mathbf{S}_j \mathbf{T} \mathbf{w} \\ \mathbf{y} \end{bmatrix} = \mathbf{x}_{\mathcal{R}_j}, \forall j, \text{ s.t. } \mathbf{y} = \mathbf{C} \mathbf{x} = \mathbf{C} \mathbf{T} \mathbf{w}. \quad (1)$$

We refer to the matrix \mathbf{C} as the L -length index code. For a given $\{\mathcal{R}_j\}$, $\{\mathbf{S}_j\}$, and \mathbf{T} , the matrix \mathbf{C} with the least number of rows L^* satisfying the condition in (1) is the optimal index code and L^* is the optimal index code length².

Examples: The problem described above is often encountered in practical scenarios such as cloud networks, multicast video-streaming and content-sharing. Since the users are connected to multiple datacenters, each user may have different subsets of the same data and require different subsets. The data could be low-dimensional due to its inherent nature (e.g., videos, images, and sensory data [12], [13]) or the usage of redundancy-inducing error correcting codes [15]. Here, the datacenters employ index codes to serve the users' requests to increase network efficiency and throughput. A similar problem is also encountered in distributed computing setups [16],

²Note that, in our proposed methodology, compression and index coding are performed in a unified framework. This helps to further simplify the receiver by relieving it of the separate decompression algorithm, reduces computational complexity, and improves overall system throughput.

distributed cognitive radio networks and satellite networks. Next, we describe these applications in more detail.

(1) Consider a wireless relay network represented in Fig. 2(a). Source nodes s_1 and s_2 wish to broadcast their data to user nodes u_1 , u_2 , and u_3 with the help of a relay node r_1 . The intensity signals transmitted by s_1 and s_2 decay with distance and the SNR deteriorates to the extent that their signals are not decodable beyond a certain radius of transmission. Consequently, in a transmission time slot t , nodes u_1 and r_1 successfully decode the data from s_1 , while nodes u_3 and r_1 successfully decode the data from s_2 , and the node u_2 receives a linear combination of the data from s_1 and s_2 . In time slot $t+1$, the relay node broadcasts a coded combination of data from s_1 and s_2 with which the nodes u_1 , u_2 , and u_3 are able to decode the data from both s_1 and s_2 . Here, the relay node comes up with an index code such that $\mathcal{R}_1 = \{2\}$, $\mathcal{R}_2 = \{1, 2\}$, $\mathcal{R}_3 = \{1\}$, $\mathbf{S}_1 = [1, 0]$ (USI), $\mathbf{S}_2 = [h_1, h_2]$ (CSI), $\mathbf{S}_3 = [0, 1]$ (USI), where h_1 and h_2 are the linear coefficients at u_2 . When the transmitted data is encoded with the same linear channel code at s_1 and s_2 , the transmitted data belongs to a low dimensional subspace. The columns of \mathbf{T} are the bases of this subspace created by the linear channel code.

(2) Consider a collaborative cognitive radio (CR) network of multiple low-cost CR c_i where $i = 1, 2, \dots, 12$, as illustrated in Fig. 2(b). Each CR senses a disjoint band of a wide spectrum and broadcasts the information to all its nearest neighbors. The data collected by all the CRs are finally fused at the fusion centers (FC) f_0 . The FC forms the complete map of the wideband spectrum. This complete map has to be conveyed back to the CRs. Since the CR network is power constrained, the FC conveys this information to its neighbors in least number of broadcasts using an index code. Further, as the low-cost CRs have limited memory, each CR stores only a linear combination of all the data it receives. Further, the CRs c_1, c_2, c_3 , and c_4 develop index codes to broadcast to their neighbors. At the FC, $\mathcal{R}_1 = \{1, 8, 9\}^c$, $\mathcal{R}_2 = \{2, 5, 7\}^c$, $\mathcal{R}_3 = \{3, 6, 10\}^c$, $\mathcal{R}_4 = \{4, 11, 12\}^c$,

$$\mathbf{S}_{1[1,8,9]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & h_8 & h_9 \end{bmatrix}, \mathbf{S}_{2[2,5,7]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & h_5 & h_7 \end{bmatrix},$$

$$\mathbf{S}_{3[3,6,10]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & h_6 & h_{10} \end{bmatrix}, \text{ and } \mathbf{S}_{4[4,11,12]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & h_{11} & h_{12} \end{bmatrix}.$$

Due to the inherent sparsity in the wideband spectrum activity, the spectrum data is low dimensional in nature [17].

(3) Consider a network of devices served by a central server of facial image databases, with each device requesting a few facial images while possessing images of other faces. Such a scenario commonly occurs in biometric verification systems and security monitoring applications. As described in the previous section, it is known that the facial image data of hundreds of pixels in dimension belong to a lower dimensional linear subspace [18]. Therefore, a subspace-aware index coding in this scenario will improve throughput, speed and scalability of the network.

(4) Consider a network of users connected to a cloud of datacenters hosting a common dataset. One such cloud network is illustrated in Fig. 2(c). The online users could be simultaneously performing operations such as document-editing or video-streaming or file-sharing. Since the users are connected to multiple datacenters, each user may contain different subsets of the same data and require different subsets. The data could belong to a low-dimensional linear subspace due to either its inherent nature (e.g., videos [19]) or the usage of redundancy-inducing error correcting codes. For multicast transmissions, the data centers employ index codes to serve the users' requests. This increases the overall network efficiency and throughput.

III. OPTIMAL INDEX CODE LENGTH

An important step towards solving the problem stated in Sec. II is to identify the minimum length of the index code. Without loss of generality, we assume $P = 1$. Let \mathbf{R}_j be a $V_j \times N$ matrix such that $\mathbf{R}_j \mathbf{x} = \mathbf{x}_{\mathcal{R}_j}$. Splitting \mathbf{D}_j into sub-matrices $\mathbf{A}_j \in \mathbb{R}^{V_j \times M_j}$ and $\mathbf{B}_j \in \mathbb{R}^{V_j \times L}$, we can write (1) as

$$\mathbf{D}_j \begin{bmatrix} \mathbf{S}_j \mathbf{T} \mathbf{w} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_j & \mathbf{B}_j \end{bmatrix} \begin{bmatrix} \mathbf{S}_j \mathbf{T} \mathbf{w} \\ \mathbf{y} \end{bmatrix} = (\mathbf{A}_j \mathbf{S}_j + \mathbf{B}_j \mathbf{C}) \mathbf{T} \mathbf{w}$$

$$\mathbf{x}_{\mathcal{R}_j} = \mathbf{R}_j \mathbf{x}, \quad \forall j. \quad (2)$$

Since $\mathbf{x} = \mathbf{T} \mathbf{w}$ and \mathbf{w} can be any arbitrary vector in \mathbb{R}^D , from (1) and (2), we can write $\mathbf{B}_j \mathbf{C} \mathbf{T} = (\mathbf{R}_j - \mathbf{A}_j \mathbf{S}_j) \mathbf{T}$, $\forall j$. This can be expressed succinctly as

$$\mathbf{B} \mathbf{C} \mathbf{T} = (\mathbf{R} - \mathbf{A} \mathbf{S}) \mathbf{T} = \tilde{\mathbf{R}} \mathbf{T}, \quad (3)$$

where

$$\begin{aligned}
\mathbf{B} &\triangleq [\mathbf{B}_1^T, \mathbf{B}_2^T, \dots, \mathbf{B}_U^T]^T \in \mathbb{R}^{(\sum_j V_j) \times L}, \\
\mathbf{S} &\triangleq [\mathbf{S}_1^T, \mathbf{S}_2^T, \dots, \mathbf{S}_U^T]^T \in \mathbb{R}^{(\sum_j M_j) \times N}, \\
\mathbf{A} &\triangleq \text{diag}([\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_U]), \\
\mathbf{R} &\triangleq [\mathbf{R}_1^T, \mathbf{R}_2^T, \dots, \mathbf{R}_U^T]^T, \\
\tilde{\mathbf{R}} &\triangleq \mathbf{R} - \mathbf{A}\mathbf{S} \in \mathbb{R}^{\sum_j V_j \times D}. \\
\mathbf{D}_j &\triangleq [\mathbf{A}_j \ \mathbf{B}_j]
\end{aligned} \tag{4}$$

Now, the optimal index code is the matrix \mathbf{C} that satisfies (3) and has the least value of $L(> 0)$. Since \mathbf{C} has only linearly independent rows, the rank of \mathbf{C} is L . Therefore, the goal is to minimize $\text{rk}(\mathbf{C})$ such that (3) is satisfied.

Note: When index coding is performed without the knowledge of the underlying subspace (we refer to this scenario as the *subspace-unaware* case) or when the data is not low-dimensional, we have $\mathbf{T} = \mathbf{I}$.

Lemma 1. $\text{rk}(\mathbf{C}\mathbf{T}) = \text{rk}(\tilde{\mathbf{R}}\mathbf{T})$.

Proof. If $\sum_j V_j < L$, then the index code length is larger than the number of data packets required. Therefore, $\sum_j V_j \geq L$; hence, $\text{rk}(\mathbf{B}) \leq L$. As governed by (2) and (3), when the decoding is successful at the receivers, $\mathbf{C}\mathbf{x} \in \text{span}(\mathbf{B})$; hence, $\text{rk}(\mathbf{B}) \geq \dim(\mathbf{C}\mathbf{x}) = L$. This proves that $\text{rk}(\mathbf{B}) = L$.

Note that by choosing the index code as $\mathbf{C} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T$ (i.e., $L = D$), and the decoder matrices as $\mathbf{B} = \mathbf{R}\mathbf{T}$ and $\mathbf{A} = \mathbf{0}$, all the required packets can be trivially decoded at the receivers. Therefore, the index code is optimal only when $L \leq D$. Now, $\text{rk}(\mathbf{C}\mathbf{T}) \leq \min(L, D) = L$, and we have

$$\text{rk}(\mathbf{B}\mathbf{C}\mathbf{T}) \leq \min(\text{rk}(\mathbf{B}), \text{rk}(\mathbf{C}\mathbf{T})) = \text{rk}(\mathbf{C}\mathbf{T}). \tag{5}$$

Further, by Sylvester's rank inequality,

$$\text{rk}(\mathbf{B}\mathbf{C}\mathbf{T}) \geq \text{rk}(\mathbf{B}) + \text{rk}(\mathbf{C}\mathbf{T}) - L = \text{rk}(\mathbf{C}\mathbf{T}). \tag{6}$$

From (3), (5) and (6), $\text{rk}(\mathbf{B}\mathbf{C}\mathbf{T}) = \text{rk}(\tilde{\mathbf{R}}\mathbf{T}) = \text{rk}(\mathbf{C}\mathbf{T})$. \square

Now, from Sylvester's rank inequality, we get

$$\text{rk}(\mathbf{C}) \leq \text{rk}(\mathbf{CT}) + N - D = \text{rk}(\tilde{\mathbf{R}}\mathbf{T}) + N - D. \quad (7)$$

Since, $N - D$ is a fixed positive value, minimizing $\text{rk}(\tilde{\mathbf{R}}\mathbf{T})$ minimizes the upperbound on $\text{rk}(\mathbf{C})$, thereby reducing $\text{rk}(\mathbf{C})$. We use this approach of minimizing $\text{rk}(\tilde{\mathbf{R}}\mathbf{T})$ to construct index codes for low-dimensional data. Further, when \mathbf{CT} has full row-rank (i.e., $\text{rk}(\mathbf{CT}) = L$), we have $\text{rk}(\mathbf{CT}) = \text{rk}(\mathbf{C})$. For subspace-unaware case, we have $\text{rk}(\mathbf{C}) = \text{rk}(\tilde{\mathbf{R}})$ [6].

A. Throughput Gain

The length of the optimal subspace-aware index codes is defined as the following

$$L^* = \min_{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_U} \text{rk}(\tilde{\mathbf{R}}\mathbf{T}). \quad (8)$$

Next, we characterize the throughput gain obtained using subspace-aware index codes.

Theorem 1. *The length of the optimal linear index code obtained for the subspace-aware case is less than or equal to the length of the optimal linear index code obtained for the subspace-unaware case.*

Proof. Let $\tilde{L} \triangleq \min_{\mathbf{A}} \text{rk}(\tilde{\mathbf{R}})$ be the optimal subspace-unaware linear index code length, and $\tilde{\mathbf{A}} \triangleq \arg \min_{\mathbf{A}} \text{rk}(\tilde{\mathbf{R}})$. Now,

$$\begin{aligned} L^* = \min_{\mathbf{A}} \text{rk}((\mathbf{R} - \mathbf{AS})\mathbf{T}) &\leq \text{rk}((\mathbf{R} - \tilde{\mathbf{A}}\mathbf{S})\mathbf{T}) \\ &\leq \min(\text{rk}(\mathbf{R} - \tilde{\mathbf{A}}\mathbf{S}), \text{rk}(\mathbf{T})) \\ &\leq \text{rk}(\mathbf{R} - \tilde{\mathbf{A}}\mathbf{S}) = \tilde{L}. \quad \square \end{aligned}$$

Corollary 1. *The length of the optimal linear index code obtained in the subspace-aware case can be bounded as*

$$\min(\tilde{L} - (N - D), 1) \leq L^* \leq \tilde{L}$$

Proof. By Sylvester's rank inequality, for any matrix \mathbf{A} ,

$$\text{rk}(\tilde{\mathbf{R}}\mathbf{T}) \geq \text{rk}(\tilde{\mathbf{R}}) + D - N \geq \tilde{L} - (N - D). \quad (9)$$

The proof follows from (9) and Theorem 1. □

IV. CONSTRUCTION OF SUBSPACE-AWARE INDEX CODES

It is well-known that the optimization problem in (8) is NP-hard. In order to solve (8), we make a practical assumption that the users can tolerate a decoding error of at most ϵ . That is,

$$\left\| \mathbf{D}_j \begin{bmatrix} \mathbf{S}_j \mathbf{x} \\ \mathbf{y} \end{bmatrix} - \mathbf{x}_{\mathcal{R}_j} \right\| \leq \epsilon, \forall j. \quad (10)$$

Note that, subspace-unaware case with USI can be seen as special cases of (8). Index codes over real field for this case has been studied previously in the literature [7]. It is known that a subspace-unaware linear index code matrix can be obtained by solving a matrix completion problem [6], [7]. However, the optimization problem in (8) is more challenging compared to the conventional matrix completion problems. This is due to the fact that an indeterminate element in \mathbf{A} affects multiple entries in the resultant $\tilde{\mathbf{R}}\mathbf{T}$ matrix in (8), which is not the case in conventional matrix completion problems. In the next subsection, we consider the design of subspace-aware/unaware index codes with CSI/USI in a unified optimization framework.

A. Construction Algorithm for Index Codes

Let $\mathbf{Z} \triangleq [\mathbf{Z}_1^T, \dots, \mathbf{Z}_U^T]^T$ be a rank r matrix and $\mathbf{Z}_j \in \mathbb{R}^{V_j \times D}$. Now, the optimization problem can be formulated as

$$\min_{\{\mathbf{Z}_j, \mathbf{A}_j\}_{j=1}^U} \sum_{j=1}^U \|\mathbf{Z}_j - (\mathbf{R}_j - \mathbf{A}_j \mathbf{S}_j) \mathbf{T}\|_F^2 = \min_{\mathbf{Z}, \{\mathbf{A}_j\}_{j=1}^U} \|\mathbf{Z} - \tilde{\mathbf{R}}\mathbf{T}\|_F^2. \quad (11)$$

We solve the optimization problem in (11) for a range of values of r and choose the minimum value of r for which the optimization was feasible (i.e., all the constraints were satisfied) as the length of the index code (L^*).

We factorize \mathbf{Z} as $\mathbf{Z} = \mathbf{X}\mathbf{Y}$, where $\mathbf{X} \in \mathbb{R}^{(\sum_j V_j) \times r}$, and $\mathbf{Y} \in \mathbb{R}^{r \times D}$. The optimization problem in (11) is not convex in \mathbf{X} , \mathbf{Y} and \mathbf{A} simultaneously; however, it is convex in \mathbf{X} (or \mathbf{Y} or \mathbf{A}) when the rest of the optimization variables are fixed. In fact, here, each of the sub-problems can be solved in a closed form. Note that, \mathbf{Z} is a rank r approximation of $\tilde{\mathbf{R}}\mathbf{T}$ (with an error of ϵ , i.e., $\|\mathbf{Z} - \tilde{\mathbf{R}}\mathbf{T}\|_F \leq \epsilon$; index codes over \mathbb{R} enable us to obtain such a rank r approximation). The steps in solving this optimization problem are listed in Algorithm 1. The alternating minimization method is guaranteed to converge to a locally optimum solution for a sufficiently large number of iterations [20].

Let $\tilde{\mathbf{Z}}$ be the matrix formed by choosing the L^* linearly independent rows of \mathbf{Z} . Now, we set $\mathbf{C}\mathbf{T} = \tilde{\mathbf{Z}}$. At every transmission instant, if the low-dimensional vector \mathbf{w} is available at the DS,

Algorithm 1 Subspace-aware Index Code Design

Require: $\{\mathbf{R}_j\}_{j=1}^U, \{\mathbf{S}_j\}_{j=1}^U, \mathbf{T}, \epsilon, t_{\max}$

- 1: **Initialize:** Random initial values
 $t = 0; \mathbf{X} \leftarrow \mathbf{X}(0); \mathbf{Y} \leftarrow \mathbf{Y}(0); \mathbf{A}_j = \mathbf{A}_j(0), \forall j$
- 2: **Update** $\{\mathbf{A}_j\}_{j=1}^U$
 $\text{Solve: } \min_{\{\mathbf{A}_j\}_{j=1}^U} \sum_{j=1}^U \|\mathbf{Z}_j(t) - (\mathbf{R}_j - \mathbf{A}_j \mathbf{S}_j) \mathbf{T}\|_F^2$
 $\mathbf{A}_j(t+1) = (\mathbf{R}_j \mathbf{T} - \mathbf{Z}_j(t))(\mathbf{S}_j \mathbf{T})^\dagger$
- 3: **Update** \mathbf{X}
 $\text{Solve: } \min_{\mathbf{X}} \|\mathbf{X} \mathbf{Y}(t) - \tilde{\mathbf{R}}(t+1) \mathbf{T}\|_F^2$
 $\mathbf{X}(t+1) = \tilde{\mathbf{R}}(t+1) \mathbf{T} \mathbf{Y}(t)^\dagger$
- 4: **Update** \mathbf{Y}
 $\text{Solve: } \min_{\mathbf{Y}} \|\mathbf{X}(t+1) \mathbf{Y} - \tilde{\mathbf{R}}(t+1) \mathbf{T}\|_F^2$
 $\mathbf{Y}(t+1) = \mathbf{X}(t+1)^\dagger \tilde{\mathbf{R}}(t+1) \mathbf{T}$
- 5: **if** $\|\mathbf{Z}(t) - \tilde{\mathbf{R}}(t) \mathbf{T}\|_F \leq \epsilon$ **or** $t = t_{\max}$ **then**
- 6: **return** $\{\mathbf{A}_j(t+1)\}_{j=1}^U$
- 7: **else**
- 8: $t \leftarrow t + 1$
- 9: **return to** Step 2
- 10: **end if**

then the matrix \mathbf{CT} can be used for index coding to generate $\mathbf{y} = \mathbf{CT}\mathbf{w}$, else the matrix \mathbf{CTT}^\dagger is used (since $\mathbf{CTT}^\dagger \mathbf{x} = \mathbf{CTT}^\dagger \mathbf{T}\mathbf{w} = \mathbf{CT}\mathbf{w} = \mathbf{C}\mathbf{x} = \mathbf{y}$).

B. Decoding Error Analysis

Theorem 2. For an index code constructed using the proposed algorithm such that $\|\mathbf{Z} - \tilde{\mathbf{R}}\mathbf{T}\|_F \leq \epsilon$, the decoding error is bounded above by ϵ .

Proof. Let $\mathbf{R}\hat{\mathbf{x}}$ be the vector decoded at the receivers. Then, the decoding error is

$$\|\mathbf{R}\mathbf{x} - \mathbf{R}\hat{\mathbf{x}}\| = \|\mathbf{R}\mathbf{T}\mathbf{w} - (\mathbf{A}\mathbf{S}\mathbf{T}\mathbf{w} + \mathbf{B}\mathbf{C}\mathbf{T}\mathbf{w})\|, \quad (12)$$

where the values of the matrices \mathbf{A} and $\mathbf{CT} (= \tilde{\mathbf{Z}})$ are obtained from Algorithm 1. We choose \mathbf{B} such that $\mathbf{Z} = \mathbf{B}\tilde{\mathbf{Z}}$. This is possible due to the following reason. Without loss of generality, we can express \mathbf{Z} as $\mathbf{Z} = [\tilde{\mathbf{Z}}^T, \bar{\mathbf{Z}}^T]^T$, where $\bar{\mathbf{Z}}$ is the matrix of $\sum_j V_j - L$ linearly dependent rows of \mathbf{Z} . Therefore, the rows of $\bar{\mathbf{Z}}$ are in the $\text{span}(\text{rows of } \tilde{\mathbf{Z}})$, i.e., $\bar{\mathbf{Z}} = \mathbf{G}\tilde{\mathbf{Z}}$ for some matrix $\mathbf{G} \in \mathbb{R}^{\sum_j V_j - L \times L}$. Hence, by choosing \mathbf{B} as $\mathbf{B} = [\mathbf{I}_L, \mathbf{G}^T]^T$, we have $\mathbf{B}\tilde{\mathbf{Z}} = \mathbf{Z}$. Further, without loss of generality, we assume $\|\mathbf{w}\|_2 \leq 1$.

Now, from (12), the decoding error can be bounded as

$$\|\mathbf{R}\mathbf{x} - \mathbf{R}\hat{\mathbf{x}}\| \leq \|\tilde{\mathbf{R}}\mathbf{T} - \mathbf{B}\tilde{\mathbf{Z}}\|_F \|\mathbf{w}\|_2 \leq \epsilon. \quad \square$$

Remark 1: The matrices $\tilde{\mathbf{Z}}$ and $\bar{\mathbf{Z}}$ can be easily obtained from \mathbf{Z} using one of the many commonly known techniques such as using QR decomposition, and $\mathbf{G} = \bar{\mathbf{Z}}\tilde{\mathbf{Z}}^\dagger$.

Remark 2: In the proposed decoding strategy, the users need not be aware of the subspace matrix \mathbf{T} for decoding. Using the matrix \mathbf{D}_j , each user can directly decode $\mathbf{x}_{\mathcal{R}_j}$.

V. NUMERICAL RESULTS

Here, we present numerical results for the proposed index code construction algorithm and analyze its performance.

A. Comparison

First, we consider the index coding problem with USI from [7]³, where $U = 4$, $\mathcal{R}_i = \{i\}$ for $i = 1, 2, 3, 4$, and

$$\mathbf{S}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{S}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$\mathbf{S}_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{S}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}.$$

Using our proposed algorithm, the $\tilde{\mathbf{R}}$ matrix obtained for $\epsilon = 10^{-10}$ is

$$\begin{bmatrix} 1 & -0.9122099 & -1.0733032 & 0 \\ -1.0962388 & 1 & 1.1765966 & 0 \\ 0 & 0.8499089 & 1 & -1.0952999 \\ -0.8506375 & 0 & 0 & 1 \end{bmatrix}.$$

The rank of this matrix is 2, which is the optimal index code length for this problem as given in [7]. We obtain the index code for this problem by choosing the linearly independent rows in the above matrix. The index code thus obtained is

$$\begin{bmatrix} 1 & -0.9122099 & -1.0733032 & 0 \\ -0.8506375 & 0 & 0 & 1 \end{bmatrix}.$$

For example, when $\mathbf{x} = [1, 1, -1, 2]^T$ is the source-data, the reconstructed values at the users, from the designed index code were $[1, 0.9999999, -1, 2]$; this gives a decoding error of $\|\mathbf{x} - \hat{\mathbf{x}}\| = 4.02 \times 10^{-14}$. For the same problem, consider the source-data to be low-dimensional with $\mathbf{T} =$

³The algorithm in [7] can solve only the subspace-unaware index coding problem with USI (conventional matrix completion problem) which is a special case of the problem we consider in this paper.

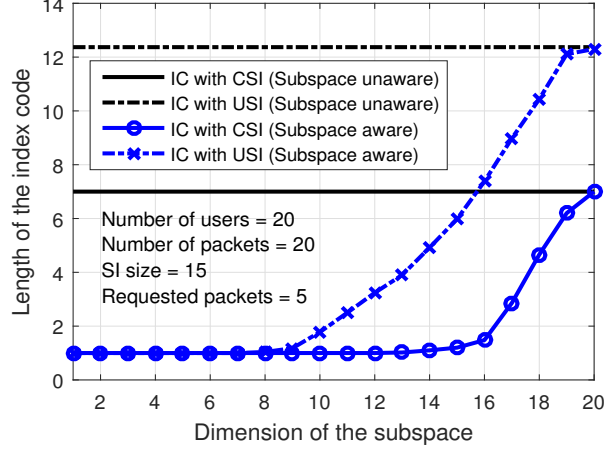


Fig. 3. Index code length obtained using the proposed algorithm for different subspace dimensions.

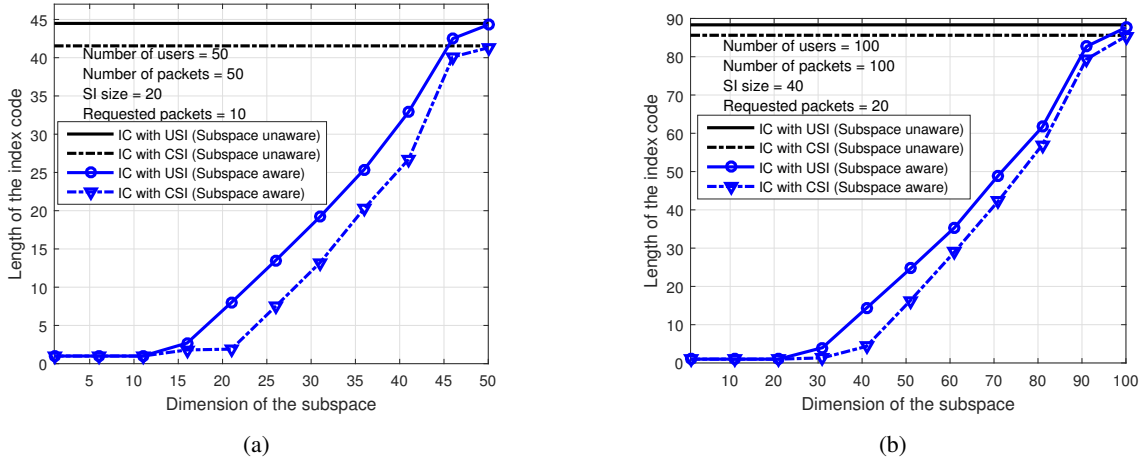


Fig. 4. Index code length obtained using the proposed algorithm for different system parameters.

$\begin{bmatrix} 1 & -2 & 1 & 1 \\ 1 & 1 & -1 & 2 \end{bmatrix}^T$. Now, $\mathbf{x} = [1, 1, -1, 2]^T = \mathbf{T}[0, 1]^T$. For this linear subspace, we get a subspace-aware index code of length 1 given by the matrix $[0, -0.3936923, 0.2644723, -0.1412844]$, and a corresponding decoding error of 8.34×10^{-14} at the users.

B. Simulation results

We simulated a simple multicast video-streaming scenario with $N = 20$ source-data packets and $U = 20$ users requesting $|\mathcal{R}_j| = 5$ data packets, each. We evaluated the index code length averaged over several instances for four different cases – namely, (1) DS is subspace-unaware and the SI is uncoded, (2) DS is subspace-unaware and the SI is coded, (3) DS is subspace-aware

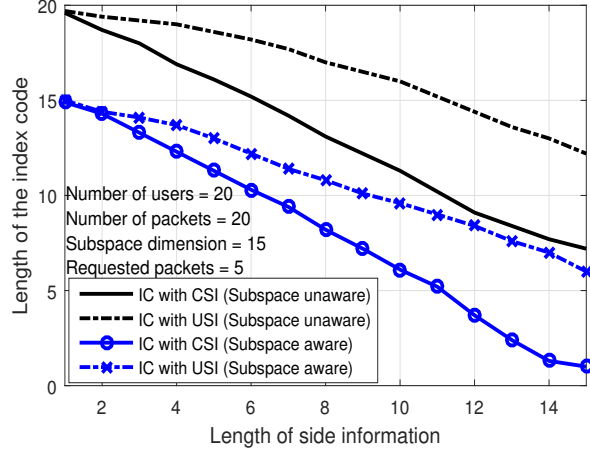


Fig. 5. Index code length obtained using the proposed algorithm for different side information sizes.

and SI is uncoded, and (4) DS is subspace-aware and the SI is coded. For fair comparison, we consider the same requirement matrix \mathbf{R} for all the cases.

In Figure 3, we plot the average index code length obtained using our proposed algorithm for different subspace dimensions fixing the SI length at each user to be $M_j = 15$. We see that when the source-data is low-dimensional, the average index code lengths obtained for the subspace-aware cases are significantly less than that of the subspace-unaware cases. The average index code lengths for the subspace-unaware cases are 7 (CSI) and 12.4 (USI). Whereas, in subspace-aware case, for $D < 9$, the average index code length is 1.1. Therefore, subspace-aware index codes reduce the transmissions required by about 91% for the USI case and by about 85% for the CSI case⁴. Also, for $9 \leq D < 20$, we observe that the subspace-aware index codes consistently outperform the subspace-unaware index codes by considerable margin.

Furthermore, from Fig. 4, we can see that even when the number of packets are 50 or 100, the subspace-aware index code outperforms the subspace-unaware index codes. For example, when the subspace dimension is half that of the number of packets (i.e., $D = 25$ when $N = 50$, and $D = 50$ when $N = 100$), subspace-aware index codes have code lengths that are 70% smaller compared to that of subspace-unaware index codes for uncoded side information, and subspace-aware index codes have 81% advantage over subspace-unaware index codes for coded side information. Thus, we can observe that irrespective of the number of packets, the subspace-

⁴ As the number of packets and users increase, the difference between the average index code length for the CSI and that of the USI case decreases.

aware index codes provide significant throughput gains over the subspace-unaware index codes.

In Figure 5, we evaluate the performance of the proposed algorithm for varying SI lengths (M_j) and fixing the subspace dimension at $D = 15$. As before, we can see that the subspace-aware index codes have significant throughput gains over the subspace-unaware index codes in both the USI and CSI cases. For instance, when $M_j = 10$, the subspace-aware cases (both USI and CSI) have an average index code length that is at least 30% smaller than that of the subspace-unaware cases.

VI. CONCLUSION

In this paper, we studied a generalization of the index coding problem that exploits source-data's structure to improve the system-throughput. We analytically characterized the length of the subspace-aware index codes and proposed an algorithm to obtain near optimal index codes. We showed that this approach significantly outperforms the conventional approaches when the source-data belongs to a low-dimensional subspace. Index coding for the case when the source-data belongs to a non-linear subspace or manifold is an interesting direction for future research. Further, network codes can also be constructed using the proposed algorithm, once the network coding problem is converted to an equivalent index coding problem [9].

REFERENCES

- [1] Y. Birk and T. Kol, "Coding on demand by an informed source (ISCOD) for efficient broadcast of different supplemental data to caching clients," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2825-2830, Jun. 2006.
- [2] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, "Index coding with side information," *IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 197-206, 2006.
- [3] M. Ji, A.M. Tulino, J. Llorca, and G. Caire, "Caching and coded multicasting: Multiple groupcast index coding," *IEEE GlobalSIP*, pp. 881-885, Dec. 2014.
- [4] V. Y. F. Tan, L. Balzano, and S. C. Draper, "Rank minimization over finite fields: Fundamental limits and coding-theoretic interpretations," *IEEE Trans. Inform. Theory*, vol. 58, no. 4, pp. 2018-2039, Apr 2011.
- [5] K. W. Shum, M. Dai, and C. W. Sung, "Broadcasting with coded side information," in *Proc. IEEE PIMRC*, 2012, pp. 89-94, Sep. 2012.
- [6] N. Lee, A. G. Dimakis, and R. W. Heath, "Index coding with coded side-information," *IEEE Commun. Letters*, vol. 19, no. 3, pp. 319-322, 2015.
- [7] X. Huang and S. El Rouayheb, "Index coding and network coding via rank minimization," *Proc. of IEEE Information Theory Workshop*, pp. 14-18, 2015.
- [8] Y. Shi and B. Mishra, "A Sparse and Low-Rank Optimization Framework for Index Coding via Riemannian Optimization," *arXiv:1604.04325*, 2016.
- [9] M. Effros, S. El Rouayheb, and M. Langberg, "An equivalence between network coding and index coding," *IEEE Trans. Inform. Theory*, vol. 61, no. 5, pp. 2478-2487, May 2015.

- [10] M. Schwartz and M. Medard, "Quasi-linear network coding," *IEEE International Symposium on Network Coding*, 2014.
- [11] S. Shintre, S. Katti, S. Jaggi, B. K. Dey, D. Katabi, and M. Medard, "Real and complex network codes: promises and challenges," *IEEE Workshop on Network Coding, Theory and Applications*, 2008.
- [12] E. Elhamifar, and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765-2781, 2013.
- [13] I. Todic, and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 27-38, 2011.
- [14] M. Mardani, G. Mateos, and G. B. Giannakis, "Subspace learning and imputation for streaming big data matrices and tensors," *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2663-2677, 2015.
- [15] S. Ubaru, A. Mazumdar, and Y. Saad, "Low rank approximation using error correcting coding matrices," *International Conference on Machine Learning*, pp. 702-710, 2015.
- [16] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," *NIPS: Workshop on Machine Learning Systems*, Dec. 2015.
- [17] Y. Ma, Y. Gao, Y. C. Liang, and S. Cui, "Reliable and efficient sub-Nyquist wideband spectrum sensing in cooperative cognitive radio networks," *IEEE Journ. on Sel. Areas in Commun.* vol. 34, no. 10, pp. 2750-2762, Oct. 2016.
- [18] A. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "Yale face database," *Center for computational Vision and Control at Yale University*, 1997, <http://cvc.yale.edu/projects/yalefaces/yalefa>.
- [19] Q. Ke, and T. Kanade, "A subspace approach to layer extraction," *Proc. of IEEE Computer Vision and Pattern Recognition*, 2001.
- [20] M. Fazel, H. Hindi, and S. Boyd, "Rank minimization and applications in system theory," *American Control Conference*, 2004.